Problem M
Milk Tea Battle

In the year 2020, to popularize computer programming amongst the wizard world, Hogwarts School of Witchcraft and Wizardry has invited professor Hanh to teach a new subject 'Computer Programming'.

To make his class more exciting, professor Hanh has brought an infinite amount of delicious milk tea to the class. During recess time, Hanh and the students will have a 'milk tea battle'.

Hanh first puts n cups on the table, and selects two constants k and x. All cups are initially empty and can store an infinite amount of milk tea. The battle consists of multiple rounds, each round happens as below:

- Hanh pours milk tea in some of the n cups. The total amount of milk tea Hanh pours does not exceed x liters. Note that the amount of milk tea Hanh put to some cup may be a non-integer number of liters.
- k students go to the stage and drink milk tea from k consecutive cups. As milk tea is super tasty, they drink all milk tea from these cups. In other words, they choose some index i $(1 \le i \le n - k + 1)$ and make all cups $i, i + 1, \ldots, i + k - 1$ empty.

In any round, Hanh can force the game to stop immediately. During this final round, only the first phase happens: Hanh does pour at most x liters of milk tea to several cups, but no student drinks milk tea. Hanh will then take a picture to upload to Facebook. To make this picture impressive, he wants at the end of the game, the **maximum** amount of milk tea in a cup to be **as large as possible**. As recess time is limited, Hanh must stop during the first $2\,000$ rounds.

Please help Hanh achieve his goal so that his photo can attract a lot of likes and loves!

Interaction

Your program first reads three integers n, k, and x, as described above. The following conditions hold for the three numbers: $1 \le n \le 50$, $1 \le k \le min(10, n)$, $1 \le x \le 1\,000$. Then in each round:

- If you want Hanh to stop the battle after this round, write `FINAL` p $(0 \le p \le n)$, where p is the number of cups you want to pour milk tea into, followed by $2 \cdot p$ numbers: c_1 v_1 c_2 $v_2 \ldots c_p$ v_p where $1 \le c_i \le n$ and $0 \le v_i \le x$, meaning you want to pour v_i liters of milk tea in the c_i-th cup. All c_i must be unique. After reading this, the interactor stops immediately and your program should not try to read anything from the standard input.
- Otherwise, write `POUR` p, followed by c_1 v_1 c_2 $v_2 \ldots c_p$ v_p, with the same meaning and constraints as above, except for the fact that Hanh will not stop the game after this round. After reading this, the interactor gives you an integer i $(1 \le i \le n - k + 1)$ indicating that students drink milk tea from cups $i, i + 1, \ldots, i + k - 1$. Your program should read this.

Note that your program should write `FINAL` within 2 000 rounds. In other words, you should write `POUR` **strictly** less than 2 000 times.

For each test case, we have a pre-calculated value OPT which we can mathematically prove that there exists a strategy for Hanh to make the maximum amount of milk tea in a cup at the end of the game at least OPT, no matter how the students play. The value OPT only depends on n, k, x and is hidden from the contestants. You are judged as passing a test case if and only if two following conditions hold:

- In every round, the total mount of poured milk tea is not greater than $(1 + 10^{-6}) \cdot x$ liters.
- At the end of the game, there exists a cup containing at least $(1 - 10^{-3}) \cdot OPT$ liters of milk tea.

Communication Example

Your output (standard output)	Kattis' answer (standard input)	Interpretation
	2 2 1	In this example, $n = 2, k = 2, x = 1$.
`POUR` 2 1 0.6 2 0.3		Hanh pours 0.6 liters of milk tea in cup 1 and 0.3 liters of milk tea in cup 2. The amount of milk tea are now 0.6 and 0.3 liters, respectively. Hanh does not stop the game after this round.
	1	Students drink all milk tea from $k = 2$ consecutive cups starting from the cup with index 1. The amount of milk tea are now 0 and 0 liters, respectively.
`FINAL` 1 2 1		Hanh pours 1 liter of milk tea in cup 2. The amount of milk tea are now 0 and 1 liters, respectively. He forces the game to stop immediately. The maximum amount of milk tea in a cup when the game ends is 1.

Notes

In this problem, the interactor is *adaptive* — the jury program can use different strategies, depending on your program's output.

When you write the solution for the interactive problem it is important to keep in mind that if you output some data it is possible that this data is first placed to some internal buffer and may be not directly transferred to the interactor. In order to avoid such situation **you have to use special 'flush' operation each time you output some data**. There are these 'flush' operations in standard libraries of almost all languages. For example, in C++ you may use `fflush(stdout)` or `cout « flush` (it depends on what do you use for output data — `scanf/printf` or `cout`). In Java you can use method 'flush' for output stream, for example, `System.out.flush()`. In Python you can use `stdout.flush()`.